

## **EXHIBIT A**

# BEST AVAILABLE COPY

Visual SourceSafe Explorer - SDG-1

File Edit View SourceSafe Tools Web Help

All projects: Contents of \$/WealthForecaster2/Engine/source Working Folder

History of \$/WealthForecaster2/Engine/source/potlimgt.cpp

History: 6 items

| Version | User    | Date            | Action                                        |
|---------|---------|-----------------|-----------------------------------------------|
| 6       | Gdovgin | 11/22/00 3:53p  | Labeled 'BUILD103_FTPSITE'                    |
| 6       | Gdovgin | 11/21/00 5:02p  | Labeled 'BUILD103'                            |
| 45      | Xwu     | 10/30/00 10:58a | Checked in \$/WealthForecaster2/Engine/source |
| 45      | Xwu     | 10/30/00 9:56a  | Checked in \$/WealthForecaster2/Engine/source |
| 44      | Iskach  | 9/28/00 7:04p   | Checked in \$/WealthForecaster2/Engine/source |
| 43      | Iskach  | 9/26/00 12:54p  | Checked in \$/WealthForecaster2/Engine/source |
| 42      | Iskach  | 9/20/00 7:37p   | Checked in \$/WealthForecaster2/Engine/source |
| 41      | Iskach  | 9/20/00 7:29p   | Checked in \$/WealthForecaster2/Engine/source |
| 6       | Gdovgin | 9/19/00 3:04p   | Labeled 'BUILD_19SEP'                         |

potlimgt.cpp:46

```

1534 double GPortfolioManager::HumanCap(int yearIndex)
1535 {
1536     double hc, yearInf;
1537
1538     int j, lex, age;
1539     double income;
1540     double mortRate;
1541     const double RealInterestRate = 0.0463; // Average 30% S&P 5000 & 70% LongTermGovBondYield-AverageInf;
1542     int k = 1 + m_pWtm->m_wtin->in_bIncludeSpouse;
1543     hc = 0;
1544
1545     for(int L=0; L<k; L++)
1546     {
1547         age = (int)CC(yearIndex, L?CTD::SPOUSE_AGE:CTD::CLIENT_AGE);
1548         lex = (int)(m_pLifeExpect[L].m_IniLifeExp+5.0+0.5)-age;
1549         yearInf = m_pWtm->m_pa->m_3DmtrSimTable[m_pWtm->m_nNRuns-1][1][yearIndex];
1550
1551         for(j=1; j<lex+6; j++)
1552         {
1553             int n = ica_min(m_pWtm->m_wtin->in_nInvestmentHorizon-1, yearIndex+j);
1554             double relInf = m_pWtm->m_pa->m_3DmtrSimTable[m_pWtm->m_nNRuns-1][1][n]/yearInf;
1555
1556             if (L == 0)
1557             {
1558                 income = CC(n, CTD::CLIENT_DC_CONTRIBUTIONS_C401KTCW)+
1559                     + CC(n, CTD::CLIENT_EMPLOYER_DC_MATCH_C401KTCW)
1560                     + CC(n, CTD::CLIENT_PROFIT_SHARING_C401KTCW)
1561                     + CC(n, CTD::CLIENT_DC_CONTRIBUTIONS_C401KYC)
1562                     + CC(n, CTD::CLIENT_EMPLOYER_DC_MATCH_C401KYC)
1563                     + CC(n, CTD::CLIENT_PROFIT_SHARING_C401KYC)
1564                     + CC(n, CTD::CLIENT_IRA_CONTRIBUTION)
1565                     + CC(n, CTD::CLIENT_ROTHIRA_CONTRIBUTION)
1566                     + CC(n, CTD::CLIENT_VA_CONTRIBUTION)
1567                     + CC(n, CTD::CLIENT_POSTTAXIRA_CONTRIBUTION)
1568                     + CC(n, CTD::CLIENT_403B_CONTRIBUTION)
1569                     + CC(n, CTD::CLIENT_457_CONTRIBUTION)
1570                     + CC(n, CTD::CLIENT_TAXABLE_CONTRIBUTION)

```

**The function, HumanCap, shown below calculates the amount of Human Capital for an investor in a given year.**

```
double GPortfolioManager::HumanCap(int yearIndex)
{
    double hc,yearInf;

    int j,lex,age;
    double income;
    double mortRate;
    const double RealInterestRate = 0.0463;//Average 30%S&p5000 & 70%
LongTermGovBondYield-AverageInflation_1970_98;
    int k = 1+m_pWtm->m_wtin->in_bIncludeSpouse;
    hc = 0;

    for(int L=0; L<k; L++)
    {
        age = (int)CC(yearIndex,L?CTD::SPOUSE_AGE:CTD::CLIENT_AGE);
        lex = (int)(m_pLifeExpect[L].m_IniLifeExp+5.0+0.5)-age;
        yearInf = m_pWtm->m_pa->m_3DmtrSimTable[m_pWtm->m_nNRuns-
1][1][yearIndex];

        for(j=1; j<lex+6; j++)
        {
            int n = ica_min(m_pWtm->m_wtin->in_nInvestmentHorizon-
1,yearIndex+j);
            double relInf = m_pWtm->m_pa->m_3DmtrSimTable[m_pWtm-
>m_nNRuns-1][1][n]/yearInf;

            if (L == 0)
            {
                income = CC(n,
CTD::CLIENT_DC_CONTRIBUTIONS_C401KTCW)+
                + CC(n,
CTD::CLIENT_EMPLOYER_DC_MATCH_C401KTCW)
                + CC(n,
CTD::CLIENT_PROFIT_SHARING_C401KTCW)
                + CC(n,
CTD::CLIENT_DC_CONTRIBUTIONS_C401KYC)
                + CC(n,
CTD::CLIENT_EMPLOYER_DC_MATCH_C401KYC)
                + CC(n,
CTD::CLIENT_PROFIT_SHARING_C401KYC)
                + CC(n, CTD::CLIENT_IRA_CONTRIBUTION)
                + CC(n,
CTD::CLIENT_ROTHIRA_CONTRIBUTION)
```

```

        + CC(n, CTD::CLIENT_VA_CONTRIBUTION)
        + CC(n,
CTD::CLIENT_POSTTAXIRA_CONTRIBUTION)
        + CC(n, CTD::CLIENT_403B_CONTRIBUTION)
        + CC(n, CTD::CLIENT_457_CONTRIBUTION)
        + CC(n,
CTD::CLIENT_TAXABLE_CONTRIBUTION)
        + CC(n,
CTD::TAXABLE_PORTFOLIO_CONTRIBUTIONS)
        + CC(n,
CTD::CLIENT_CONTRIBUTIONS_401KAT_TOTAL)
        + CC(n, CTD::CLIENT_LOAN_PAYMENT)
        + (m_pWtm->m_wtin-
>in_bIncludeClientPension?CC(n,CTD::CLIENT_PENSION):0.0)
        + (m_pWtm->m_wtin-
>in_bIncludeClientSS?CC(n,CTD::CLIENT_SS):0.0);
    }
    else
    {
        income = CC(n,CTD::SPOUSE_DC_CONTRIBUTIONS)
        +
CC(n,CTD::SPOUSE_EMPLOYER_DC_MATCH)
        + CC(n,CTD::SPOUSE_PROFIT_SHARING)
        + CC(n,CTD::SPOUSE_IRA_CONTRIBUTION)
        +
CC(n,CTD::SPOUSE_ROTHIRA_CONTRIBUTION)
        + CC(n,CTD::SPOUSE_VA_CONTRIBUTION)
        +
CC(n,CTD::SPOUSE_POSTTAXIRA_CONTRIBUTION)
        + CC(n,
CTD::SPOUSE_TAXABLE_CONTRIBUTION)
        +
CC(n,CTD::SPOUSE_CONTRIBUTIONS_401KAT_TOTAL)
        + (m_pWtm->m_wtin-
>in_bIncludeSpousePension?CC(n, CTD::SPOUSE_PENSION):0.0)
        + (m_pWtm->m_wtin-
>in_bIncludeSpouseSS?CC(n, CTD::SPOUSE_SS):0.0);
    }

    if (j+age <= MaxTableAge)
        mortRate = m_pLifeExpect[L].m_SurviveRate[j+age];
    else
        mortRate =
m_pLifeExpect[L].m_SurviveRate[MaxTableAge]*
        pow(0.5,j+age-
MaxTableAge);

```

```
        double p = pow(1+RealInterestRate,j);  
        hc += (income/relInf) * mortRate / p;  
    }  
}  
  
return hc;  
}
```